UNITED STATES PATENT AND TRADEMARK OFFICE

Patent Application for an invention entitled

IMAGE PROCESSING SYSTEM FOR DYNAMIC SUPPRESSION OF AIRBAGS USING
MULTIPLE MODEL LIKELIHOODS TO INFER THREE DIMENSIONAL INFORMATION

By:

Michael Edward Farmer
5119 Lake Bluff Road
West Bloomfield, MI 48323

Prepared by:

Michael B. Stewart, Reg. No. 36,018
Christopher J. Falkowski, Reg. No. 45,989
Rader Fishman & Grauer, PLLC
39533 Woodward Avenue, Suite 140
Bloomfield Hills, Michigan 48304
(248) 594-0600

IMAGE PROCESSING SYSTEM FOR DYNAMIC SUPPRESSION OF AIRBAGS USING MULTIPLE MODEL LIKELIHOODS TO INFER THREE DIMENSIONAL INFORMATION

BACKGROUND OF THE INVENTION

[0001] The present invention relates in general to systems used to determine whether the deployment of an airbag should be prevented or modified due to the proximity of the occupant to the airbag. In particular, the present invention is an image processing system that receives a stream of two-dimensional images and applies iterative and probability-weighted processes to infer three-dimensional characteristics to those images, and predict whether the upper torso of the occupant would be within the At-Risk-Zone ("ARZ") by the time that an air bag would actually deploy.

[0002] Conventional airbag deployment systems have contributed significantly to the safety of occupants in automobile crashes. However, there may be occasions when due to the physical proximity of an occupant to an airbag, that the deployment of an airbag may not be desirable. Airbag disablement systems are known in the art. Some of those systems attempt to process the distance between the occupant and the airbag. However, there are several significant problems with that existing art.

[0003] First, such systems require highly expensive cameras. Timing is critical to any system used to determine whether or not an airbag should be deploy. A standard video camera operates at a frequency between 50-100hz and captures between 50 to 100 image frames per second of operation. Effective airbag determinations require more frequent updates, of approximately 200 updates per second (200hz). Moreover, it would be desirable for an image processing system to predict the occupant's position in advance instead of merely identifying the occupant's location at the time that the image is captured. It would also be desirable if accurate predictions could be generated at a faster rate than the camera speed so that a less expansive standard video camera could be used instead of a more expensive highly specialized high-speed camera.

[0004] Second, prior art systems are highly susceptible to "noise" because prior art systems focus solely on the most recent image, and ignore the series of images captured mere fractions of a second earlier. "Noise" results from several factors,

including the inherent imperfections of the segmentation process which is the process of extracting a segmented image, an image of the occupant, in isolation from the surrounding area, from the ambient image, an image of the occupant and an image of the surrounding area. It would be desirable for an image processing system to utilize an iterative process that would integrate the information contained in the most recent image into a comprehensive framework that includes prior predictions and indirectly, the prior images used to make those prior predictions. It would also be desirable for all predictions to be weighted predictions. Moreover, it would be helpful for such weighted predictions to include probabilities associated with predefined occupant states such as leaning left towards the driver, leaning right away from the driver, or sitting upright, and predefined occupant modes such as crash, stationary, or human.

[0005] The third problem with prior art systems is that they rely on two-dimensional images. The images captured by cameras, including video cameras, are inherently two dimensional images. It would be useful if three-dimensional information could be inferred from a series of two dimensional images. Moreover, it would be helpful if predefined occupant states were incorporated into the iterative process of deriving a three-dimensional information from a series of two-dimensional images.

SUMMARY OF THE INVENTION

[0006] This invention relates to an image processing system used to determine whether the proximity of an occupant of a seat to an airbag is sufficiently close such that the airbag deployment system should be either disabled or modified. In particular, the present invention is an image processing system that incorporates current measurements with past predictions in order to determine the location of an occupant relative to the position of an airbag deployment system.

[0007] Two iterative multiple model Kalman filters are used to incorporate a measurement of the most recent image into the ongoing series of past predictions and image measurements relating to the position and shape of the occupant.

[0008] A camera or some other sensor is used to capture an ambient image, the image of the occupant and the area surrounding the occupant. A segmented image, the image of the occupant with the surrounding area removed from the image, is obtained through a process called segmentation. The segmented image is then subjected to an shape fitting process that fits an upper shape to the upper torso of the occupant. A

lower shape is fitted from the occupant's hips down to the occupant's feet. The two shapes overlap at one point, the centroid, in the area of the occupant's hips. In the preferred embodiment, an ellipse is the shape used.

[0009] Tracking and predicting processes are performed using multiple model Kalman filters. The process involves two parallel and simultaneous sub-processes. One process tracks important information relating to the shape of the occupant image, and other tracks important information relating to the motion of the occupant image.

[0010] The shape tracker and predictor process tracks and predicts the shape of the occupant by tracking and predicting (collectively "shape variables"): the "major axis" of the ellipse, a bisecting line through the longer cross section of the ellipse; the "minor axis" of the ellipse, a bisecting line through the shorter cross section of the ellipse; and the y-coordinate or "height" of the centroid in the segmented image. For each shape variable, the velocity and acceleration, in addition to position, are each tracked and predicted.

[0011] An occupant's shape is defined into one of several predefined "states" such as leaning towards the driver ("left"), leaning away from the driver ("right"), and sitting upright ("center"). All processing by the shape tracker is modeled in each of the predefined states. Using the probabilities associated with potential transitions from one state to another state, and incorporating past predictions through the use of an iterative and weighted Kalman filter, the shape tracker and predictor will determine which state the occupant is in by determining which model ellipse most closely resembles the most recently measured ellipse. Thus, even though the tilt angle towards or away from the driver (the "sideways tilt angle" Φ), is perpendicular to the image plane and thus is not visible in a two-dimensional image, the system can infer valuable three-dimensional information about a two-dimensional image.

[0012] The determination of shape state is then used to update the estimate for all shape variables. A shape estimate is updated using incoming measurements, state predictions, and the Kalman filter gain matrices. The shape tracker and predictor can be used to generate estimates more frequently than a camera can capture images. Thus, the system does not require the use of a high-speed video camera.

[0013] The motion tracker and predictor processes information relating to the position, velocity, and acceleration of the occupant. "Motion variables" include the x-

-3-

coordinate of the centroid which is used to measure the "distance" between the x-coordinate and the airbag and the "forward title angle" θ, the degree to which the occupant is tilted towards the front of the vehicle.

[0014] Position, velocity, and acceleration are tracked and predicted for all motion variables. An occupant's motion is defined into one of several predefined "modes" such as "stationary," "human," or "crash". All processing by the motion tracker and predictor is modeled in each of the predefined modes. Predictions are generated for each mode. Using the probabilities associated with potential transitions from one mode to another mode, and incorporating past predictions thought the use of an iterative and weighted Kalman filter, the motion tracker and predictor will generate one combined estimate for each motion variable. The motion tracker and predictor can be used to generate estimates more frequently than a camera can capture images. Thus, the system does not require the use of a high-speed video camera.

[0015] The final step is the actual determination of whether or not the occupant is within an At-Risk-Zone ("ARZ"). An ARZ Intrusion Indicator determines whether the upper torso of the occupant would be within the ARZ at the time in which an airbag would deploy. The ARZ Intrusion Indicator also determines whether the person is leaning so far towards the driver that they miss the ARZ. For different vehicle models, the ARZ will be set differently. This calculation is updated very quickly using the outputted state transition matrix which allows the system to predict ahead the position and shape at a rate much more quickly than the rate at which the camera collects data.

[0016] Various aspects of this invention will become apparent to those skilled in the art from the following detailed description of the preferred embodiment, when read in light of the accompanying drawings.

BRIEF DESCRIPTION OF THE DRAWINGS

[0017] Fig. 1 shows a partial view of the surrounding environment for one potential embodiment of the invention.

[0018] Fig. 2 shows a high-level process flow of the use of the image processing system.

[0019] Fig. 3 shows a high level block diagram of an image processing system.

[0020] Fig 4 shows the application of the ellipse fitting process.

[0021] Fig. 5 show an upper torso ellipse with all corresponding shape variables and motion variables (excluding velocity and acceleration derivations).

[0022] Fig. 6 shows three potential shape states and three potential sideways tilt angles.

[0023] Fig. 7 shows a markov chain of probabilities relating to the three shape states of left, right, and center.

[0024] Fig. 8 shows a markov chain of probabilities relating to the three motion modes of human, stationary, and crash.

[0025] Fig. 9 shows the detailed process flow for determining the shape of the occupant.

[0026] Fig. 10 shows the detailed process flow for determining the motion of the occupant.

DETAILED DESCRIPTION OF THE PREFERRED EMBODIMENT

A. PARTIAL VIEW OF SURROUNDING ENVIRONMENT

[0027] Referring now to the drawings, illustrated in Fig. 1 is a partial view of the surrounding environment for one potential embodiment of the invention., an image processing system 16. If an occupant 18 is present, the occupant 18 sits on a seat 20. In the preferred embodiment, a camera or other sensor 22 is attached in a roof liner 24, above the occupant 18 and closer to a front windshield 26 than the occupant 18. The camera 22 should be placed in a slightly downward angle towards the occupant 18 in order to capture changes in the angle of the occupant's 18 upper torso resulting from forward or backward movement in the seat 20. There are or course many potential locations for a camera 22 that are well known in the prior art. Similarly, a computer system 30 could be located virtually anywhere on a vehicle. In the preferred embodiment, the computer system 30 is located near the camera 22 to avoid sending camera images through long wires. An airbag controller 32 is shown in an instrument panel 34, although the present invention could still function even if the airbag deployment system were located in a different environment. Similarly, an airbag deployment system 36 is also located in the instrument panel 34 in front of the occupant 18 and the seat 20.

B.   HIGH LEVEL PROCESS FLOW

[0028] Fig. 2 discloses a very basic and high level process flow surrounding the use of the image processing system 16. An ambient image 38 of a seat area 21 including both the occupant 18 and the seat area 21 is captured by the video camera 22 or any other sensor capable of rapidly capturing a series of images. In the figure, the seat area 21 includes the entire occupant, although under some circumstances and embodiments, only a portion of the occupant's 18 image will be captured, particularly if the camera 22 is positioned in a location where the lower extremities may not be viewable. The ambient image 38 is sent to the computer system 30. The computer system 30 determines whether or not the occupant 18 would be in as At-Risk-Zone ("ARZ") at the time of an airbag deployment. The ARZ determination is sent to the airbag controller 32 which in turn controls the airbag deployment system 36.

C.   COMPUTER SYSTEM PROCESS FLOW

[0029] Fig. 3 gives a high level process flow as to the inner workings of the computer system 30. The ambient image 38 of the occupant 18 and surrounding seat area 21 is an input of the computer system 30. A segmentation routine 40 is performed which extracts a segmented image 42 of the occupant, an image which includes only the occupant and not the surrounding seat area 21, from the ambient image 38. The segmentation process 40 is well known under existing art. Segmentation 40 is accomplished electronically through the use of software, and thus segmentation could be performed in a computer separate from the computer system in 30.

[0030] The segmented image 42 of the occupant is then subject to an ellipse fitting routine 44. An upper ellipse is fitted around the upper torso of the occupant 18. The upper torso includes everything from the occupant's 18 head down to the occupant's hips. A lower ellipse is fitted from the occupant's 18 toes up to the occupant's 18 hips. Both ellipses overlap at one point which is referred to as the centroid. The output of the ellipse fitting routine 44 will be discussed in greater detail below along with the discussion relating to Figs. 4, 5, and 6. The process and practice of generating an upper ellipse, a lower ellipse, and a centroid to represent different parts of the occupant 18 is well known in the art. In the preferred embodiment of the invention, the ellipse fitting routine 44 is applied by the computer system 30, but in

other embodiments, the ellipse fitting routine 44 could be performed by a separate computer from the computer system at 30.

[0031] Information relating to the upper ellipse, lower ellipse, and centroid are then sent to a tracking and predicting system 46. The tracking and predicting system 46 applies distinct sub-processes simultaneously with each other. A shape tracker 48 tracks variables relating to the "shape" of the upper ellipse. This process is described in greater detail below in the discussion relating to Fig. 9. A motion tracker 50 tracks variables relating to the "motion" of the upper ellipse, specifically with respect the horizontal distance between the occupant 18 and the airbag deployment system 36 in the instrument panel 34. This process is described in greater detail below in the discussion relating to Fig. 10.

[0032] Every embodiment of the computer system 30 needs a tracking and predicting system 46 track and predict the position, velocity, and acceleration of certain key measurements relating to shape and motion of the upper torso of the occupant 18 using a multiple model and iterative process, which incorporates the latest measurement into a series of ongoing measurements and predictions. In the preferred embodiment of the invention, the tracking and predicting system 46 incorporates what is known in the prior art as a Kalman filter. A Kalman filter is very powerful in that it incorporates past predictions and measurements in a weighted fashion, supports estimations of past, present, and even future states, and it can do so even when the precise nature of the modeled system is unknown. An academic paper entitled "An Introduction to the Kalman Filter" by Greg Welch and Gary Bishop is attached and incorporated by reference. The general equation for the Kalman filter is as follows:

$$X_{(new\ prediction)} = X_{(old\ prediction)} + Gain[-X_{(old\ prediction)} + X_{(measured)}]$$

In a Kalman filter, "Gain" represents the perceived accuracy of the most recent measurement. A Gain of 0 indicates such a poor measurement that it is of no value, and thus the new estimate $X_{(new\ estimate)}$ is simply the value of the old estimate $X_{(old\ estimate)}$.

$$X_{(new\ estimate)} = X_{(old\ estimate)} + 0[-X_{(old\ estimate)} + X_{(measured)}]$$

$$X_{(new\ estimate)} = X_{(old\ estimate)} + 0$$

$$X_{(new\ estimate)} = X_{(old\ estimate)}$$

A Gain of 1 indicates such confidence in the most recent measurement $X_{(measured)}$ that the new prediction $X_{(new\ estimate)}$ is simply the value of the most recent measurement $X_{(measured)}$.

$$X_{(new\ estimate)} = X_{(old\ estimate)} + 1[-X_{(old\ estimate)} + X_{(measured)}]$$

$$X_{(new\ estimate)} = X_{(old\ estimate)} - X_{(old\ estimate)} + X_{(measured)}]$$

$$X_{(new\ estimate)} = X_{(measured)}$$

In a real world application, the Gain is virtually always greater than 0 and less than 1. The Gain thus determines to what degree a new measurement can change the previous aggregate estimate or prediction of the location of an object, in the case of the instant invention, the occupant 18 is the object being tracked. Both the shape tracker 48 and the motion tracker 50 are described in greater detail below, along with Figs. 9 and 10 respectively.

[0033] The output from both the shape tracker 48 and the motion tracker 50 are sent to an At-Risk-Zone ("ARZ") Intrusion Indicator 52. The ARZ Intrusion Indicator 52 determines the proximity of the closest point of the upper ellipse relative to the instrument panel 34, the distance which defines the ARZ. The ARZ Intrusion Indicator 52 also determines whether or not the occupant 18 is leaning so far toward the driver that the occupant 18 will miss the ARZ altogether. Calculations in the ARZ Intrusion Indicator 52 are done more quickly than the rate in which the camera 22 collects data. The ARZ Intrusion Indicator 52 is specifically configured for the particular model of vehicle. Thus, a minivan will typically have a different ARZ than a compact automobile. The ARZ Intrusion Indicator is discussed in greater detail below, along with the discussion relating to Fig. 5.

[0034] The determination by the ARZ Intrusion Indicator 52 is sent to the airbag controller 32 so that the airbag controller 32 can implement the appropriate response with the airbag deployment system 36. The deployment of the airbag at 36 can either be totally disabled or merely modified as a result of the ARZ determination made at 52.

D. ELLIPSE FITTING ROUTINE

[0035] Fig. 4 illustrates the ellipse fitting routine 44 implemented by the computer system 30. The upper ellipse 58 extends from the hips up to the head of the occupant 18. The lower ellipse 56. extends down from the hips to include the feet of the occupant 18. If the entire area from an occupant's 18 hips down to the occupant's 18 feet is not visible, a lower ellipse is generated to represent what is visible. Both ellipses overlap at a point known as the centroid 54. In non-preferred embodiments, shapes other than ellipses are used to represent the upper and lower parts of an occupant 18. The ellipse fitting routine is well known in the art. The image processing system does not utilize the lower ellipse 56, and is only used to generate the centroid 54.

[0036] Fig. 5 illustrates many of the characteristics that can be inputted into the computer system 30 from the ellipse fitting routine 44. Motion characteristics include the x-coordinate ("distance") 60 of the centroid 54 and the forward tilt angle ("θ") 62. Shape measurements include the y-coordinate ("height") 58 of the centroid 54, the length of the major axis of the ellipse ("major") 64 and the length of the minor axis of the ellipse ("minor") 66. Rate of change information, such as velocity and acceleration, are also captured for all shape and motion measurements, so in the preferred embodiment of the invention there are nine shape characteristics (height, height, height, major, major, major, minor, minor, and minor) and six motion characteristics (distance, distance, distance, θ, θ', and θ''). The sideways tilt angle Φ is not shown because it is perpendicular to the image plane, and this the sideways title angle Φ is derived, not measured, as discussed in greater detail below.

[0037] Fig. 6 illustrates the sideways tilt angle "(Φ") 73. In the preferred embodiment of the invention, there are only three shape states, leaning left towards the driver (left) 76, sitting upright (center) 72, and leaning right away from the driver (right) 74, with tilt sideways tilt angles of $-\Phi$, 0, and Φ. In the preferred embodiment, Φ is set at a value between 15 and 40 degrees, depending on the nature of the vehicle being used.

E.   MARKOV PROBABILITY CHAINS

[0038] The present invention is directed to a multiple-model probability weighted implementation of a Kalman filter for all shape characteristics and all motion characteristics. Each shape characteristic has a separate Kalman filter equation for each shape state. Similarly, each motion characteristic has a separate Kalman filter equation for each motion mode. In every embodiment of the invention, the occupant 18 has at least one shape state and at least one motion mode. There are certain predefined probabilities associated with a transition from one state to another state. These probabilities can best be illustrated through the use of Markov chains.

[0039] Fig. 7 illustrates the three shape states used in the preferred embodiment of the invention. In the preferred embodiment, an occupant 18 is either leaning towards the driver ("left") 80, sitting upright ("center") 78, or leaning away from the driver ("right") 82. The probability of an occupant being in a particular state and then ending in a particular state can be identified by lines originating at a particular shape state with arrows pointing towards the subsequent shape state. For example, the probability of an occupant in center state remaining in center state P^{C-C} is represented by the arrow at 84. The probability of moving from center to left P^{C-L} is represented by the arrow 92 and the probability of moving from center to right P^{C-R} is 98. The total probabilities resulting from an initial state of center 78 must add up to 1.

$$P^{C-C} + P^{C-L} + P^{C-R} = 1.0$$

Furthermore, all of the probabilities originating from any particular state must also add up to 1.0.

[0040] The arrow at 86 represents the probability that a left tilting occupant 18 will sit centered P^{L-C}, by the next interval of time. Similarly, the arrow at 96 represents the probability that a left tilting occupant will tilt right P^{L-R} by the next interval of time, and the arrow at 90 represents the probability that a left tilting occupant will remain tilting to the left P^{L-L}. The sum of all possible probabilities originating from an initial tilt state of left must equal 1.

$$P^{L-C} + P^{L-L} + P^{L-R} = 1.0$$

[0041] Lastly, the arrow at 100 represents the probability that a right tilting occupant will remain tilting to the right P^{R-R}, the arrow at 88 represents the probability that a right tilting occupant will enter a centered state P^{R-C}, and the arrow at 94 represents

the probability that an occupant will tilt towards the left P^{R-L}. The sum of all possible probabilities originating from an initial tilt state of right equals 1.

$$P^{R-C} + P^{R-L} + P^{R-R} = 1.0$$

[0042] As a practical matter, the typical video camera 22 used in the preferred embodiment captures between 50 to 100 frames each second. Thus, it is essentially impossible for a left 80 leaning occupant to become a right 82 leaning occupant, or for a right 82 leaning occupant to become a left 80 leaning occupant, in a mere 1/50 of a second. It is far more likely that a left 80 leaning occupant will first enter a center state 78 before becoming a right 82 leaning occupant, and similarly, it is far more realistic for a left 80 leaning occupant to become a centered 78 occupant before becoming a right 82 leaning occupant. Thus, in the preferred embodiment of, P^{L-R} at 96 is always set at zero and P^{R-L} at 94 will also always be set at zero. The three probability equations relating to shape state are thus as follows:

$$P^{C-C} + P^{C-L} + P^{C-R} = 1.0$$
$$P^{R-C} + P^{R-R} = 1.0$$
$$P^{L-C} + P^{L-L} = 1.0$$

[0043] Fig. 8 illustrates a similar Markov chain to represent the relevant probabilities relating to motion modes. The preferred embodiment of the invention uses three motion modes: stationary 102, represents a human occupant 18 in a mode of stillness, such as while asleep; human 112, represents a occupant 18 behaving as a typical passenger in an automobile or other vehicle, one that is moving as a matter of course, but not in an extreme way; and crash 122, represents the occupant 18 of a vehicle that is in a mode of crashing.

[0044] The probability of an occupant being in a particular state and then ending in a particular state can be identified by lines originating in the current state with arrows pointing to the new state. For example, the probability of an occupant in a stationary state remaining in stationary state P^{S-S} is represented by the arrow at 108. The probability of moving from stationary to human P^{S-H} is represented by the arrow 114 and the probability of moving from stationary to crash P^{S-C} is 90. The total probabilities resulting from an initial state of stationary 102 must add up to 1.

$$P^{S-S} + P^{S-H} + P^{S-C} = 1.0$$

[0045] Similarly, the probability of human to human is P^{H-H} at 116, human to stationary is P^{H-S} at 112, and human to crash is P^{H-C} at 120, and stationary to crash P^{H-C} is 90. The total probabilities resulting from an initial state of stationary 102 must add up to 1.

$$P^{H-H} + P^{H-C} + P^{H-C} = 1.0$$

[0046] Lastly, the probability of going from crash to crash is P^{C-C} at 124, crash to stationary is P^{C-S} at 110, and crash to human is P^{C-H} at 118. The total probabilities resulting from an initial state of crash 122 must add up to 1.

$$P^{C-C} + P^{C-S} + P^{C-H} = 1.0$$

[0047] As a practical matter, it is highly unlikely for an occupant 18 to ever leave the state of crash at 122 once that state has been entered. Under most scenarios, a crash at 122 ends the trip for the occupant 18. Thus, in the preferred embodiment, P^{C-H} is set to zero and P^{C-S} is also set to zero. The three equations for motion mode probabilities in the preferred embodiment are as follows:

$$P^{C-C} = 1.0$$
$$P^{H-H} + P^{H-C} + P^{H-S} = 1.0$$
$$P^{S-C} + P^{S-H} + P^{S-S} = 1.0$$

[0048] The transition probabilities associated with the various shape states and motion modes are used to generate a Kalman filter equation for each combination of characteristic and state. The results of those filters can then be aggregated in to one result, using the various probabilities to give the appropriate weight to each Kalman filter. All of the probabilities are predefined by the user of the invention.

[0049] The markov chain probabilities provide a means to weigh the various Kalman filters for each characteristic and for each state and each mode. The tracking and predicting system 46 incorporates the markov chain probabilities in the form of two subsystems, the shape tracker and predictor 48 and the motion tracker and predictor 50.

F.   SHAPE TRACKER AND PREDICTOR

[0050] Fig. 9 discloses a detailed flow chart for the shape tracker and predictor 48. In the preferred embodiment of the invention, the shape tracker and predictor 48 tracks and predicts the major axis of the upper ellipse ("major"), the minor axis of the upper

ellipse ("minor"), and the y-coordinate of the centroid ("height"). Each characteristic has a vector describing position, velocity, and acceleration information for the particular characteristic. The major vector is [major, major , major], with major representing the rate of change in the major or velocity and major representing the rate of change in major velocity or acceleration. Accordingly, the minor vector is [minor, minor , minor], and the height vector is [height, height , height].   Any other shape vectors will similarly have position, velocity, and acceleration components. The first step in the shape tracking and prediction process is an update of the shape prediction at 126.

1.   Update Shape Prediction

[0051] An update shape prediction process is performed at 126. This process takes the last shape estimate and extrapolates that estimate into a future prediction using a transition matrix.

*Updated Vector Prediction = Transition Matrix * Last Vector Estimate*

The transition matrix applies Newtonian mechanics to the last vector estimate, projecting forward a prediction of where the occupant 18 will be on the basis of its past position, velocity, and acceleration. The last vector estimate is produced at 130 as described below. The process from 126 to 128, from 128 to 130, and from 130 to 132, loops back to 126. The process at 126 requires that an estimate be previously generated at 130, so processing at 126 and 128 is not invoked the first time through the repeating loop that is steps 126 – 132.

[0052] The following equation is then applied for all shape variables and for all shape states, where x is the shape variable,   t represents change over time (velocity), and ½ t^2 represents acceleration:

$$\text{Updated Vector Prediction} = \begin{pmatrix} 1 & t & \tfrac{1}{2}t^2 \\ 0 & 1 & t \\ 0 & 0 & 1 \end{pmatrix} * \begin{pmatrix} x \\ x \\ x \end{pmatrix}$$

In the preferred embodiment of the invention, there would be nine updated vector predictions at 134 because there are three shape states and three non-derived·shape variables in the preferred embodiment, and 3 x 3 = 9.

[0053] The updated shape vector predictions are:

Updated major for center state.

Updated major for right state.

Updated major for left state.

Updated minor for center state.

Updated minor for right state.

Updated minor for left state.

Updated height for center state.

Updated height for right state.

Updated height for left state.

2.   Update Covariance and Gain Matrices

[0054] After the shape predictions are updated for all variables and all states at 126, the shape prediction covariance matrices, shape gain matrices, and shape estimate covariance matrices must be updated at 128. The shape prediction covariance accounts for error in the prediction process. The gain, as described above, represents the weight that the most recent measurement is to receive and accounts for errors in the measurement segmentation process. The shape estimate covariance accounts for error in the estimation process.

[0055] The prediction covariance is updated first. The equation to be used to update each shape prediction covariance matrix is as follows:

Shape Prediction Covariance Matrix =

[State Transition Matrix * Old Estimate Covariance Matrix * transpose(State Transition Matrix)] + System Noise

The state transition matrix is the matrix that embodies Newtonian mechanics used above to update the shape prediction. The old estimate covariance matrix is generated from the previous loop at 130. On the first loop from 126 through 132, step 128 is skipped. Taking the transpose of a matrix is simply the switching of rows with columns and columns with rows, and is known under the art. Thus, the transpose of the state transition matrix is the state transition matrix with the rows as columns and the columns as rows. System noise is a matrix of constants used to incorporate the idea of noise in the system. The constants used in the system noise matrix are set by

the user of the invention, but the practice of selecting noise constants are known in the art.

[0056] The next matrix to be updated is the gain matrix. As discussed above, the gain represents the confidence of weight that a new measurement should be given. A gain of one indicates the most accurate of measurements, where past estimates may be ignored. A gain of zero indicates the least accurate of measurements, where the most recent measurement is to be ignored and the user of the invention is to rely solely on the past estimate instead. The role played by gain is evidenced in the basic Kalman filter equation where

$$X_{(new\ estimate)} = X_{(old\ estimate)} + Gain[-X_{(old\ estimate)} + X_{(measured)}]$$

[0057] The gain is not simply one number because one gain exists for each combination of shape variable and shape state. The general equation for updating the gain is as follows:

Gain =

Shape Prediction Covariance Matrix * transpose(Measure Matrix) * inv(Residue Covariance)

The shape covariance matrix is calculated above. The measure matrix is simply a way of isolating and extracting the position component of a shape vector while ignoring the velocity and acceleration components for the purposes of determining the gain. The transpose of the measure matrix is simply [1 0 0]. The reason for isolating the position component of a shape variable is because velocity and acceleration are actually derived components, only position can be measured by a snapshot. Gain is concerned with the weight that should be attributed to the actual measurement.

[0058] In the general representation of a Kalman filter, $X_{(new\ estimate)} = X_{(old\ estimate)} + Gain[-X_{(old\ estimate)} + X_{(measured)}]$, the residue represents the difference between the old estimate and the new measurement. There are entire matrices of residue covariances. The inverse of the residue covariance matrix is used to update the gain matrix. It is known in the art how to take the inverse of a matrix, which is a simple linear algebra process. The equation for residue covariance matrix is as follows:

Residue Covariance =

[Measurement Matrix * Prediction Covariance * transpose(Measurement Matrix)] + Measurement Noise

The measurement matrix is a simple matrix used to isolate the position component of a shape vector from the velocity and acceleration components. The prediction covariance is calculated above. The transpose of the measurement matrix is simply a one row matrix of [1 0 0] instead of a one column matrix with the same values. Measurement noise is a constant used to incorporate error associated with the sensor 22 and the segmentation process 40.

[0059] The last matrix to be updated is the shape estimate covariance matrix, which represents estimation error. As estimations are based on current measurements and past predictions, the estimate error will generally be less substantial than prediction error. The equation for updating the shape estimation covariance matrix is as follows:

Shape Estimate Covariance Matrix =

(Identity Matrix – Gain Matrix * Measurement Matrix) * Shape Predictor Covariance Matrix

[0060] An identity matrix is known in the art, and consists merely of a diagonal line of 1's going from top left to bottom right, with zeros at every other location. The gain matrix is computed and described above. The measure matrix is also described above, and is used to isolate the position component of a shape vector from the velocity and acceleration components. The predictor covariance matrix is also computed and described above.

3. Update Shape Estimate

[0061] An update shape estimate process is invoked at 138. The first step in this process is to compute the residue.

Residue = Measurement – (Measurement Matrix * Prediction Covariance)

Then the shape states themselves are updated.

Updated Shape Vector Estimate = Shape Vector Prediction +(Gain * Residue)

When broken down into individual equations, the results are as follows:

$$X^C_{(major\ at\ t)} = X^C_{(major\ at\ t)} + Gain[-X^C_{(major\ at\ t-1)} + X^C_{(measured\ major)}]$$

$$X^L_{(major\ at\ t)} = X^L_{(major\ at\ t)} + Gain[-X^L_{(major\ at\ t-1)} + X^L_{(measured\ major)}]$$

$$X^R_{(major\ at\ t)} = X^R_{(major\ at\ t)} + Gain[-X^R_{(major\ at\ t-1)} + X^R_{(measured\ major)}]$$

$$X^C_{(minor\ at\ t)} = X^C_{(minor\ at\ t)} + Gain[-X^C_{(minor\ at\ t-1)} + X^C_{(measured\ minor)}]$$

$$X^L_{(minor\ at\ t)} = X^L_{(minor\ at\ t)} + Gain[-X^L_{(minor\ at\ t-1)} + X^L_{(measured\ minor)}]$$

$$X^R_{(minor\ at\ t)} = X^R_{(minor\ at\ t)} + Gain[-X^R_{(minor\ at\ t-1)} + X^R_{(measured\ minor)}$$

$$X^C_{(height\ at\ t)} = X^C_{(height\ at\ t)} + Gain[-X^C_{(height\ at\ t-1)} + X^C_{(measured\ height)}]$$

$$X^L_{(height\ at\ t)} = X^L_{(height\ at\ t)} + Gain[-X^L_{(height\ at\ t-1)} + X^L_{(measured\ height)}]$$

$$X^R_{(height\ at\ t)} = X^R_{(height\ at\ t)} + Gain[-X^R_{(height\ at\ t-1)} + X^R_{(measured\ height)}]$$

In the preferred embodiment, C represents the state of center, L represents the state of leaning left towards the driver, and R represents the state of leaning right away from the driver.

4.   Generate Combined Shape Estimate

[0062] The last step in the repeating loop between steps 126 and steps 132 is a generate combined shape estimate step at 132. The first part of that process is to assign a probability to each shape vector estimate. The residue covariance is re-calculated, using the same formula as discussed above.

Covariance Residue Matrix =

[Measurement Matrix * Prediction Covariance Matrix * transpose(Measurement Matrix)] + Measurement Noise

[0063] Next, the actual likelihood for each shape vector is calculated. The system 16 determines which state the occupant is in by comparing the predicted values for the various states with the recent best estimate of what the current values for the shape variables actually are.

$$Likelihood \begin{pmatrix} C \\ R \\ L \end{pmatrix} = e^{-(residue-offset)^2/2\sigma^2}$$

There is no offset in the preferred embodiment of the invention because it is assumed that offsets cancel each other out, and that the system's 16 processes. Sigma represents variance, and is defined in the implementation phase of the invention by a human developer. It is known in the art how to assign a useful value for sigma by looking at data.

[0064] The state with the highest likelihood determines the sideways tilt angle Φ. If the occupant 18 is in a centered state, the sideways tilt angle is 0 degrees. If the occupant 18 is tilting left, then the sideways tilt angle is $-\Phi$. If the occupant 18 is tilting towards the right, the sideways tilt angle is Φ. In the preferred embodiment of

the invention, Φ and $- \Phi$ are predefined on the basis of the type and model of vehicle using the system 16.

[0065] Next, state probabilities are updated from the likelihood generated above and the pre-defined markovian mode probabilities discussed above..

$$P^C = P^{C-C} + P^{R-C} + P^{L-C}$$

$$P^R = P^{R-R} + P^{C-R}$$

$$P^L = P^{L-L} + P^{C-L}$$

The equations for the updated mode probabilities are as follows, where L represents the likelihood of a particular mode as calculated above:

Probability of mode Left =

$$1/[L^L * (P^{L-L} + P^{C-L}) + L^R * (P^{R-R} + P^{C-R}) + L^C * (P^{C-C} + P^{R-C} + P^{L-C})] * L^L * (P^{L-L} + P^{C-L})$$

Probability of mode Right =

$$1/[L^L * (P^{L-L} + P^{C-L}) + L^R * (P^{R-R} + P^{C-R}) + L^C * (P^{C-C} + P^{R-C} + P^{L-C})] * L^R * (P^{R-R} + P^{C-R})$$

Probability of mode Center =

$$1/[L^L * (P^{L-L} + P^{C-L}) + L^R * (P^{R-R} + P^{C-R}) + L^C * (P^{C-C} + P^{R-C} + P^{L-C})] * L^C * (P^{C-C} + P^{L-C} + P^{L-C})$$

[0066] The combined shape estimate is ultimately calculated by using each of the above probabilities, in conjunction with the various shape vector estimates.

$$X = \text{Probability of mode Left} * X^{Left}$$
$$+ \text{Probability of mode Right} * X^{Right}$$
$$+ \text{Probability of mode Center} * X^{Center}$$

X is any of the shape variables, including a velocity or acceleration derivation of a measure value.

G. MOTION TRACKER AND PREDICTOR

[0067] The motion tracker and predictor 50 in Fig. 10 functions similarly in many respects, to the shape tracker and predictor 48 in Fig. 9. The motion tracker and predictor 50 tracks different characteristics and vectors than the shape tracker. In the preferred embodiment of the invention, the x-coordinate of the centroid and the forward tilt angle θ ("θ"), and their corresponding velocities and accelerations (collectively "motion variables") are tracked and predicted. The x-coordinate of the centroid is used to determine the distance between the occupant 18 and a location

within the automobile such as the instrument panel 34, the airbag deployment system 36, or some other location in the automobile. In the preferred embodiment, the instrument panel 34 is used since that is where the airbag is generally deployed from.

[0068] The x-coordinate vector includes a position component (x), a velocity component (x), and an acceleration component (x). The θ vector similarly includes a position component (θ), a velocity component (θ), and an acceleration component (θ). Any other motion vectors will similarly have position, velocity, and acceleration components.

1. Update Motion Prediction

[0069] An update motion prediction process is performed at 134. This process takes the last motion estimate and extrapolates that estimate into a future prediction using a transition matrix.

Updated Vector Prediction = Transition Matrix * Last Vector Estimate

The transition matrix applies Newtonian mechanics to the last vector estimate, projecting forward a prediction of where the occupant 18 will be on the basis of its past position, velocity, and acceleration. The last vector estimate is produced at 138 as described below. The process from 134 to 136, from 136 to 138, and from 138 to 140, loops back to 134. The process at 134 requires that an estimate be previously generated at 138, so processing at 134 and 136 is not invoked the first time through the repeating loop that is steps 134 – 140.

[0070] The following equation is then applied for all motion variables and for all motion modes:

$$
\text{Updated Vector Prediction} =
\begin{pmatrix} 1 & t & \tfrac{1}{2}\,t^2 \\ 0 & 1 & t \\ 0 & 0 & 1 \end{pmatrix}
*
\begin{pmatrix} x \\ x \\ x \end{pmatrix}
$$

In the preferred embodiment of the invention, there would be six updated vector predictions at 134 because there are three motion modes and two motion variables in the preferred embodiment, and 3 x 2 = 6. The updated motion predictions are:

   Updated x-coordinate for crash mode.

   Updated x-coordinate for human mode.

   Updated x-coordinate for stationary mode.

   Updated θ for crash mode.

Updated θ for human mode.

Updated θ for stationary mode.

2.　　Update Covariance and Gain Matrices

[0071] After the motion predictions are updated for all motion variables and all modes at 134, the motion prediction covariance matrices, motion gain matrices, and motion estimate covariance matrices must be updated at 136. The motion prediction covariance accounts for error in the prediction process. The gain, as described above, represents the weight that the most recent measurement is to receive and accounts for errors in the measurement and segmentation process. The motion estimate covariance accounts for error in the estimation process.

[0072] The prediction covariance is updated first. The equation to be used to update each motion prediction covariance matrix is as follows:

Motion Prediction Covariance Matrix =

State Transition Matrix * Old Estimate Covariance Matrix * transpose(State Transition Matrix) + System Noise

The state transition matrix is the matrix that embodies Newtonian mechanics used above to update the motion prediction. The old estimate covariance matrix is generated from the previous loop at 136. On the first loop from 134 through 140, step 134 is skipped. Taking the transpose of a matrix is simply the switching of rows with columns and columns with rows, and is known under the art. Thus, the transpose of the state transition matrix is the state transition matrix with the rows as columns and the columns as rows. System noise is a matrix of constants used to incorporate the idea of noise in the system. The constants used in the system noise matrix are set by the user of the invention, but the practice of selecting such constants are known in the art.

[0073] The next matrix to be updated is the gain matrix. As discussed above, the gain represents the confidence of weight that a new measurement should be given. A gain of one indicates the most accurate of measurements, where past estimates may be ignored. A gain of zero indicates the least accurate of measurements, where the most recent measurement is to be ignored and the user of the invention is to rely on the past

estimate instead. The role played by gain is evidenced in the basic Kalman filter equation where

$$X_{(new\ estimate)} = X_{(old\ estimate)} + Gain[-X_{(old\ estimate)} + X_{(measured)}]$$

[0074] The gain is not simply one number but an entire matrix because one gain exists for each combination of motion variable and motion mode. The general equation for updating the gain is as follows:

Gain =

*Motion Prediction Covariance Matrix * transpose(Measure Matrix) * inv(Residue Covariance)*

The motion covariance matrix is calculated above. The measure matrix is simply a way of isolating and extracting the position component of a motion vector while ignoring the velocity and acceleration components for the purposes of determining the gain. The transpose of the measure matrix is simply [1 0 0]. The reason for isolating the position component of a motion variable is because velocity and acceleration are actually derived components. Position is the only component actually measured, and because gain is concerned with the weight that should be attributed to the actual measurement, derived variables should be isolated.

[0075] In the general representation of a Kalman filter, $X_{(new\ estimate)} = X_{(old\ estimate)} + Gain[-X_{(old\ estimate)} + X_{(measured)}]$, the residue represents the difference between the old estimate and the new measurement. There are entire matrices of residue covariances. The inverse of the residue covariance matrix is used to update the gain matrix. It is known in the art how to take the inverse of a matrix, which is a simple linear algebra process. The equation for residue covariance matrix is as follows:

Residue Covariance =

*[Measurement Matrix * Prediction Covariance * transpose(Measurement Matrix)] + Measurement Noise*

The measurement matrix is a simple matrix used to isolate the position component of a motion vector from the velocity and acceleration components. The prediction covariance is calculated above. The transpose of the measurement matrix is simply a one row matrix of [1 0 0] instead of a one column matrix with the same values. Measurement noise is a constant used to incorporate error associated with the sensor 22 and the segmentation process 40.

[0076] The last matrix to be updated is the motion estimate covariance matrix, which represents estimation error. As estimations are based on current measurements and past predictions, the estimate error will generally be less substantial than the prediction error. The equation for updating the motion estimation covariance matrix is as follows:

Motion Estimate Covariance Matrix =

(Identity Matrix – Gain Matrix * Measurement Matrix) * Motion Predictor Covariance Matrix

[0077] An identity matrix is known in the art, and consists merely of a diagonal line of 1's going from top left to bottom right, with zeros at every other location. The gain matrix is computed and described above. The measure matrix is also described above, and is used to isolate the position component of a motion vector from the velocity and acceleration components. The predictor covariance matrix is also computed and described above.

3. Update Motion Estimate

[0078] An update motion estimate process is invoked at 138. The first step in this process is to compute the residue.

Residue = Measurement – (Measurement Matrix * Prediction Covariance)

Then the motion states themselves are updated.

Motion Vector Estimate =Motion Vector Prediction + (Gain* Residue)

When broken down into individual equations, the results are as follows:

$$X^H_{\text{(x-coordinate at t)}}=X^H_{\text{(x-coordinate at t)}}+\text{Gain}[-X^H_{\text{(x-coordinate at t-1)}}+X^H_{\text{(measured x-coordinate)}}]$$

$$X^S_{\text{(x-coordinate at t)}}=X^S_{\text{(x-coordinate at t)}}+\text{Gain}[-X^S_{\text{(x-coordinate at t-1)}}+X^S_{\text{(measured x-coordinate)}}]$$

$$X^C_{\text{(x-coordinate at t)}}=X^C_{\text{(x-coordinate at t)}}+\text{Gain}[-X^C_{\text{(x-coordinate at t-1)}}+X^C_{\text{(measured x-coordinate)}}]$$

$$X^H_{(\theta \text{ at t})}=X^H_{(\theta \text{ at t})}+\text{Gain}[-X^H_{(\theta \text{ at t-1.})}+X^H_{\text{(measured }\theta)}]$$

$$X^S_{(\theta \text{ at t})}=X^S_{(\theta \text{ at t})}+\text{Gain}[-X^S_{(\theta \text{ at t-1)}}+X^S_{\text{(measured }\theta)}]$$

$$X^C_{(\theta \text{ at t})}=X^C_{(\theta \text{ at t-1)}}+\text{Gain}[-X^C_{(\theta \text{ at t-1)}}+X^C_{\text{(measured }\theta)}]$$

In the preferred embodiment, H represents the mode of human, C represents the mode of crash, and S represents the mode of stationary.

4. Generate Combined Motion Estimate

[0079] The last step in the repeating loop between steps 134 and steps 140 is a generate combined motion estimate step at 140. The first part of that process is to

assign a probability to each motion vector estimate. The residue covariance is re-calculated, using the same formula as discussed above.

Covariance Residue Matrix =

[Measurement Matrix * Prediction Covariance Matrix * transpose(Measurement Matrix)] + Measurement Noise

[0080] Next, the actual likelihood for each motion vector is calculated.

$$\text{Likelihood}\begin{pmatrix} C \\ H \\ S \end{pmatrix} = e^{-(\text{residue} - \text{offset})^2 / 2\sigma^2}$$

There is no offset in the preferred embodiment of the invention because it is assumed that offsets cancel each other out, and that the system's 16 processes. Sigma represents variance, is defined in the implementation phase of the invention by a human developer. It is known in the art how to assign a useful value for sigma by looking at data.

[0081] Next, mode probabilities are updated from the likelihood generated above and the pre-defined markovian mode probabilities discussed above..

$$P^C = P^{C\text{-}C} + P^{S\text{-}C} + P^{H\text{-}C}$$

$$P^H = P^{H\text{-}H} + P^{S\text{-}H}$$

$$P^S = P^{S\text{-}S} + P^{H\text{-}S}$$

The equations for the updated mode probabilities are as follows, where L represents the likelihood of a particular mode as calculated above:

Probability of mode Stationary =

$$1/[L^S * (P^{S\text{-}S} + P^{H\text{-}S}) + L^H * (P^{H\text{-}H} + P^{S\text{-}H}) + L^C * (P^{C\text{-}C} + P^{S\text{-}C} + P^{H\text{-}C})] * L^S * (P^{S\text{-}S} + P^{H\text{-}S})$$

Probability of mode Human =

$$1/[L^S * (P^{S\text{-}S} + P^{H\text{-}S}) + L^H * (P^{H\text{-}H} + P^{S\text{-}H}) + L^C * (P^{C\text{-}C} + P^{S\text{-}C} + P^{H\text{-}C})] * L^H * (P^{H\text{-}H} + P^{S\text{-}H})$$

Probability of mode Human =

$$1/[L^S * (P^{S\text{-}S} + P^{H\text{-}S}) + L^H * (P^{H\text{-}H} + P^{S\text{-}H}) + L^C * (P^{C\text{-}C} + P^{S\text{-}C} + P^{H\text{-}C})] * L^C * (P^{C\text{-}C} + P^{S\text{-}C} + P^{H\text{-}C})$$

[0082] The combined motion estimate is ultimately calculated by using each of the above probabilities, in conjunction with the various motion vector estimates.

$$X = \text{Probability of mode Human} * X^{Human}$$
$$+ \text{Probability of mode Crash} * X^{Crash}$$
$$+ \text{Probability of mode Stationary} * X^{Stationary}$$

X is any of the motion variables, including a velocity or acceleration derivation.

H.   ARZ INTRUSION INDICATOR

[0083] The ARZ Intrusion Indicator at 52 takes the outputs of both the shape indicator at 26 and the motion tracker at 28 and determines if the occupant is within the At-Risk-Zone ("ARZ"), an area predefined to be too close to the airbag to allow safe deployment of the airbag. This calculation can utilize any of the motion or shape variables, or even characteristics that are not shape or motion variables. In the preferred embodiment, the ARZ Intrusion Indicator 52 determines whether or not the closest point on the upper ellipse 58 is within the predefined ARZ. In the preferred embodiment, the ARZ is defined taking into consideration the specific environment presented by a particular type of automobile. The ARZ Intrusion Indicator 52 also determines whether or not the occupant 18 is leaning so far to the driver side of the automobile, that the occupant 18 is not within the ARZ. ARZ calculations are updated very quickly using the outputted state transition matrix which allows the system 16 to predict the position and shape in advance, and at a rate more quickly than the rate in which the sensor 22 collects data.

[0084] In accordance with the provisions of the patent statutes, the principles and modes of operation of this invention have been explained and illustrated in preferred embodiments. However, it must be understood that this invention may be practiced otherwise than is specifically explained and illustrated without departing from its spirit or scope.